



# **On Understanding How to Introduce an Innovation to an Open Source Project**

Christopher Oezbek and Lutz Prechelt

Institut für Informatik

Freie Universität Berlin

05/21/2007

- You have an software engineering invention:
  - A new bug tracker
  - A new release process
  - A new code generator
  - A new IDE plug-in
- Empirical evaluation necessary. Possible approaches:
  - Industry trial
  - Laboratory trial
  - **OSS project trial**

# Necessary steps

1. Select suitable target projects
2. Approach the project
3. Achieve adoption of invention
4. Phase out
5. Observe invention use

# 1. Select a suitable project

- Use Theoretical / Purposeful sampling
- Project needs right characteristics to make introduction possible, viable and interesting:
  - Open source development style (not only license)
    - distributed, public communication & archival, basic process patterns established, open for outsiders to join
  - 5-15 active developers
  - Basic affinity for change
    - Avoid highly mature projects unless your invention targets the maintenance phase

# Necessary steps

1. Select suitable target projects
2. Approach the project
3. Achieve adoption of invention
4. Phase out involvement
5. Observe invention use

## 2. Approaching the project I

- Important concepts:
  - Gift culture
    - Beware: Not the effort for you to create the gift counts, but the usefulness and mental compatibility for the project.
  - Onion model
- Prepare by lurking and learning about the project
- Offering without joining?
  - If you need recognition in the project, it might be necessary first to join conventionally
  - Zero-acquaintance approach needs to engage responsibility in addressee and counteract skepticism [Rossner 07]

# Who to approach?

- Maintainer, Leaders
- Whole Project
- Individual Members
- Proposition: Correlate with type of innovation decision [Rogers 03]
  - Authority innovation decisions => Maintainers, Leaders
  - Collective innovation decisions => Whole Project
  - Optional innovation decisions => Individual members

# Necessary steps

1. Select suitable target projects
2. Approach the project
3. Achieve adoption of invention
4. Phase out involvement
5. Observe invention use

### 3. Achieve adoption of invention

- Proposition: Pay attention to benefit structure of innovation
  - Necessary to consider both the individual and project level
  - Only large advantage or social pressures can make a non-beneficial task be picked up by someone
  - Consequence: The researcher/introducer has to pay these cost initially

# Necessary steps

1. Select suitable target projects
2. Approach the project
3. Achieve adoption of invention
4. Phase out involvement
5. Observe invention use

## 4. Phasing out

- Ethically imperative to perform graceful and gradual exit to not harm the project.
- On success, phase-out as innovation becomes self-sustaining
  - Be prepared to be available for questions regarding innovation for some continued time (6-12 months)
- On failure, need to revert damage to infrastructure.
- Recruit members for the project to offset loss of researcher leaving.

# Necessary steps

1. Select suitable target projects
2. Approach the project
3. Achieve adoption of invention
4. Phase out
5. ~~Observe invention use~~

# The Introduction of Inventions

- Motivation:
  - To enable the improvement of OSS projects by interested stakeholders beyond code contribution
- Research method:
  - Case studies of innovation introduction
    - Role based process improvement (information manager) and a well known tool (JUnit) so far
  - I follow the same procedure presented but from a slightly different angle

# Your opinion?

## Gathering insights...

- As with all empirical work: Preparation is key
- Primary source will be the interaction with the project
  - Keep diary of conversation and important events
  - Keep snapshots of project (not all technology has history functions)
- Keep track of innovation adoption/use within project
  - For quantitative approaches you need to start collecting data before introduction starts.
- Define in advance what an successful introduction is
  - Adoption by XX% of developers
  - Increase of baseline metrics of project
  - Positive attitude towards innovation gathered via survey

# Ethically Adequate?

- Key Considerations (Belmont Report):
  - Maintain the autonomy of project members
  - Provide Beneficence
  - Justice
- Unproblematic?
  - OSS projects highly autonomous and in control.
  - Public setting and perceived as such
  - Low harm/risk and relatively immediate benefits to project.
  - Avoid collecting personal information but rather focus on technology.
  - Make sure that any initial burden is carried by researcher.
- Literature: [Ethical decision-making and Internet research: Recommendations from the AOIR ethics working committee](#)

- Lots of frameworks to use for modeling interaction:
  - theory of fields
  - network-actor theory
  - personal foundational practices
    - Sensing, Envisioning, Offering, Executing, Adopting, Sustaining, Leading
- Reinvention occurs a lot
  - Acceptance of the invention but in ways unintended by you as the inventor

# Case I: Information management

- Dedicated information manager as a role based process improvement
  - Gathers decisions, recurring issues from mailing-list and generates coherent documents
- Wiki used as an underlying technology
- Introduced in 4 OSS projects
  - First case used to establish general structure of innovation process, later 3 to explore types of approaches
- Lots of reinvention occurred
  - Surprisingly: Decisions almost never occur
  - representation, making complicated procedures (how to release) available for project members, community building / organization

## Case II: JUnit

- Well established technology, optional innov. decision
- Research question:
  - For projects that already have adequate JUnit coverage: How did the introduction occur?
  - For those without: Introduce JUnit with a focus on "Which situations / arguments cause members to start using the invention?"
- Results so far (keep in mind that JUnit is well-known):
  - Leaders of the project use testing as a keyword to emphasize a desire for quality without there being activity to base the progress of the project in test cases.
  - Newbies to the project embrace unit testing as a entry-path. Writing tests has little chance conflicting with somebody else.
  - General knowledge about unit testing is low.

# What is an innovation?

- *Innovation is the adoption of a new practice by a group* [Denning & Dunham 06]
- The group in the context of this paper is the developer community of an OSS project.
- Examples
  - Linux Kernel Patch Process
  - KDE Feature Freeze Cycle
  - Using Trac as an integrated wiki, repository and bugtracker
- What Innovations are not:
  - Product features (Preemptive scheduling in the kernel)
  - Inventions (have not yet been adopted)

# External validity of insights?

- Do innovations that are successful within an OSS project transfer to corporate settings?
  - Management championship / extrinsic benefits can spur adoption
  - Full time developer will see economies of scale for most innovations.
  - OSS developers are known for being critical of external influence.
- Limitations:
  - Unsuccessful innovation adoption does not necessarily mean inadequacy of the innovation for corporate settings because of OSS particular restriction.
  - Introduction/penetration speed will be lower for most OSS projects